



UNIVERSIDADE DO MINHO
DEPARTAMENTO DE ENGENHARIA E RECURSOS DO MAR

CURSO DE LICENCIATURA EM INFORMÁTICA DE GESTÃO

RELATÓRIO DE PROJETO DE LICENCIATURA
ANO LETIVO 2017/2018 – 4º ANO

Autor: Gideão Gil Lopes Rocha, N.º 2843

Mindelo, 2018

Gideão Gil Lopes Rocha

**PHARM GO - APLICAÇÃO MÓVEL E
PLATAFORMA WEB – INTERMEDIACÃO COM
FARMÁCIAS**

Universidade do Mindelo

Mindelo, Julho de 2018

Gideão Gil Lopes Rocha

PHARM GO - APLICAÇÃO MÓVEL E PLATAFORMA WEB – INTERMEDIACÃO COM FARMÁCIAS

Trabalho de Conclusão de Curso apresentado
à **Universidade do Minho** como parte dos
requisitos para obtenção do grau de
Licenciatura em Informática de gestão

Orientador:

Doutor João Dias

Mindelo, 2018

Dedico este trabalho a minha mãe e mais ninguém,
não há quem me suporte assim como ela e o faz
tão bem, todo amor retribuído nunca chegará para
pagar os seus feitos.

AGRADECIMENTOS

É de agradecer a Deus e a minha mãe Deolinda Lopes, os quais devo fazer uma vénia porque sem ela nunca tinha chegado até aqui, e nunca poderei retribuir tanto carinho.

Também agradeço aos meus colegas/amigos/parceiros/irmãos de curso, pois, durante os 4 anos a parceria foi total, o companheirismo foi sincero e a luta foi conjunta.

Também agradeço a minha namorada, que agora é esposa, que com todo apoio e carinho me incentivou a superar os meus medos e alcançar as minhas metas com todo o louvor.

Finalmente agradeço aos meus docentes que sem eles, sem a sua pressão, seu apoio, seu carinho, seus resmungos e suas chatices não tinha chegado até aqui.

A todos as pessoas que direta ou indiretamente contribuíram para o meu sucesso durante este percurso um muito obrigado e que Deus vos abençoe!

“Como gerente você é pago para estar desconfortável. Se você está confortável, é um sinal seguro de que você está fazendo as coisas erradas.”

Peter Drucker

RESUMO

No âmbito das inovações tecnológicas, o homem procura acompanhar essas inovações para poder estar de igual com o mercado em competição ou mesmo tentar acompanhá-lo. Em Cabo Verde, segundo a Agência Nacional das Comunicações, nas últimas décadas, o sector das telecomunicações tem vindo a assumir um papel determinante para o desenvolvimento económico, o crescimento da produtividade e a difusão tecnológica, daí a importância da compilação de dados que constituem indicadores que permitem conhecer o estado atual da infraestrutura de telecomunicações de Cabo Verde, nomeadamente ao nível de Telefonia fixa e móvel, internet e televisão por assinatura. Mostrando, assim, o enorme crescimento da utilização de dispositivos móveis na vida quotidiana dos cabo-verdianos. Os mesmos têm procurado as aplicações existentes nesses dispositivos, de forma a melhorar a sua qualidade de vida e diminuindo cada vez mais o seu esforço em vista a gestão da sua agenda. Assim, sendo isso de extrema importância, é imprescindível a busca pelo conhecimento para a criação de plataformas modernos, eficientes e habilidosos. Portanto, o presente trabalho tem como objetivo desenvolver um aplicativo baseado na localização de farmácias chamado “Pharm Go”. Onde o utilizador pode procurar uma farmácia aberta e o aplicativo retorna com a farmácia e o produto procurado. O aplicativo foi desenvolvido para plataforma Android, utilizando GPS, com o apoio de uma plataforma Web em PHP e HTML com acesso ao banco de dados MySQL.

Palavras-chave: Tecnologia, Plataformas, Aplicativo, Android, GPS, PHP, HTML e MySQL.

ABSTRACT

In the scope of technological innovations, the man seeks to follow these innovations in order to be equal with the market in competition or even try to accompany him. In Cape Verde, according to the National Communications Agency, in the last decades the telecommunications sector has assumed a decisive role for economic development, productivity growth and technological diffusion, hence the importance of compiling data that are indicators which allow to know the current state of the telecommunications infrastructure of Cape Verde, namely the level of fixed and mobile telephony, internet and pay-TV. This shows the tremendous growth of the use of mobile devices in the daily lives of Cape Verdeans. They have been looking for the existing applications in these devices, in order to improve their quality of life and increasingly less effort in order to manage their agenda. Thus, being of the utmost importance, the search for knowledge for the creation of modern, efficient and skillful platforms is essential. Therefore, the present work aims to develop an application based on the location of pharmacies called "Pharm Go". Where the user can search for an open pharmacy and the application returns with the pharmacy and the product sought. The application was developed for Android platform, using GPS, with the support of a Web platform in PHP and HTML with access to the MySQL database.

Keywords: Technology, Platforms, Applications, Android, GPS, PHP, HTML e MySQL.

Sumário

1. Introdução	1
1.1 Contextualização	1
1.2. Motivação	3
1.3. Objetivos	4
1.3.1. Objetivos Gerais	4
1.3.2. Objetivos Específicos	4
1.4. Metodologia	4
1.5. Estrutura do Trabalho	5
2. Enquadramento Teórico	6
2.1. Computação Móvel	6
2.1.1. Dispositivos para Computação Móvel	7
2.1.2. Tipos de Aplicativos Móveis	8
2.1.2.1. Aplicativo Nativo	8
2.1.2.2. Aplicativo Web	8
2.1.2.3. Aplicativo Híbrido	9
2.1.3. Principais Plataformas de Desenvolvimento	9
2.2. Android	10
2.3. IOS	10
2.4. Windows Phone	10
2.5. O Android	11
2.5.1. Arquitetura do Android	11
2.5.2. Applications	12
2.5.3. Applications Framework	12
2.5.4. Bibliotecas e Android Runtime	12

2.5.5. Linux Kernel	13
2.6. Componentes de uma Aplicação	13
2.6.1. Activities	14
2.6.2. Services	14
2.6.3. Content Providers	14
2.6.4. Broadcast Receiver	14
2.6.5. Android Manifest	15
2.7. Ciclo de Vida de uma Aplicação	15
3. Análise do Sistema a Desenvolver	18
3.1. Visão Geral do Sistema	18
3.2. Requisitos do Sistema	18
3.2.1. Requisitos Funcionais da Aplicação	19
3.2.2. Requisitos Não Funcionais	20
3.3. Modelagem do Sistema	21
3.3.1. Diagramas de Caso de Uso	21
3.3.2. Diagrama de Classes	22
3.3.3. Diagrama de Sequência	22
3.3.4. Diagrama de Atividade	23
3.3.5. Base de Dados	23
4. Ferramentas e Tecnologias Utilizadas	24
4.1. Android SDK	24
4.2. Linguagem PHP	25
4.3. Base de Dados MySQL	25
4.4. JSON	26
4.5. Genymotion	26

4.6. HTML	27
4.6.1. Versão HTML5	28
4.7. Sistema de Posicionamento Global	28
5. Implementação do Protótipo	29
5.1. Configuração da Aplicação	30
5.2. Criando a Base de Dados Local	32
5.3. Pesquisando Produto	33
5.4. Seguindo a Fila	34
5.5. Ver Informações da Farmácia	35
5.6. Localização no Mapa	36
5.7. Área Reservada apenas para Farmacêuticos	37
5.8. Envio de Email	39
5.9. Website	40
6. Conclusão	41
6.1. Resultados Obtidos	41
6.2. Trabalhos Futuros	42
7. Referências Bibliográficas	43

LISTA DE FIGURAS

Figura 1: Arquitetura plataforma Android	11
Figura 2: Componente de um aplicativo Android	13
Figura 3: Ciclo de vida de um aplicativo Android	16
Figura 4: Arquitetura do sistema	18
Figura 5: Diagrama de Caso de Uso	22
Figura 6: Diagrama de Atividade	23
Figura 7: Android Manifest	31
Figura 8: Base de dados Pharm Go	32
Figuras 9 e 10: Funcionalidade Pesquisar	33
Figuras 11 e 12: Funcionalidade Fila	34
Figuras 13 e 14: Funcionalidade Informações	35
Figuras 15 e 16: Funcionalidade Mapa	36
Figuras 17, 18 e 19: Funcionalidade Login, Adicionar e Listar	37
Figuras 20, 21 e 22: Funcionalidade Listar, Atualizar e Eliminar	38
Figuras 23 e 24: Funcionalidade Email	39
Figuras 25: Website Login	40
Figuras 26: Página Inicial	40

LISTA DE SIGLAS E ABREVIATURAS

ADT - Android Development Tools

API - Application Programming Interface

APP - Aplicativo

GPS - Sistema de Posicionamento Global

HTML - Hypertext Markup Language

HTTP - HyperText Transfer Protocol

IDE - Integrated Development Environment

JSON - JavaScript Object Notation

OS - Operational System

PHP - Personal Home Page

SDK - Software Development Kit

SGBD - Sistema de Gerenciamento de Banco de Dados

SQL - Structured Query Language

UML - Unified Modeling Language

1. Introdução

1.1 Contextualização

Com as inovações tecnológicas têm se notado que o homem procura sempre atualizar as suas técnicas e/ou ferramentas, nas áreas de dispositivos móveis, de forma a ter eficiência, eficácia, melhorar a sua qualidade de vida com o menor esforço possível.

Segundo pesquisas, mais de 3 bilhões de pessoas têm um telefone telemóvel, e o mercado de aplicações móveis transformou-se numa área muito viável, rendendo bilhões todos os anos. E há que fazer uma análise de duas formas.

Os utilizadores comuns atualmente buscam um telemóvel com um visual sofisticado, moderno, de fácil navegação, assim como uma infinidade de aplicações e novos recursos. Tanto as empresas quanto os desenvolvedores buscam uma plataforma moderna e ágil para o desenvolvimento das suas aplicações. Os fabricantes como a LG, Motorola, Samsung, HTC, Intel, Sony e algumas outras, precisam de uma plataforma robusta e com muitas funcionalidades para lançar no mercado os seus produtos.

Em Cabo Verde, segundo a Agência Nacional das Comunicações, nas últimas décadas, o sector das telecomunicações tem vindo a assumir um papel determinante para o desenvolvimento económico, o crescimento da produtividade e a difusão tecnológica, daí a importância da compilação desses dados que constituem indicadores que permitem conhecer o estado atual da infraestrutura de telecomunicações do país, nomeadamente ao nível de Telefonia fixa e móvel, internet e televisão por assinatura.

Como principais resultados, destacam-se:

- O mercado de telefonia fixa, mantém a tendência decrescente do número de assinantes, apresentando um decréscimo de 0,13% em relação ao mesmo período do ano anterior.
- No serviço de Telefonia móvel, em Dezembro de 2015 constavam 646.334 cartões SIM ativos no mercado móvel em Cabo Verde, o que representa um crescimento de cerca de 5,4% em relação ao mesmo período de 2014. Fixando a taxa de penetração neste período (até Dezembro de 2015) em 123%.
- Em relação às assinaturas do serviço de acesso à internet, em 2015 registou-se 370.187, dos quais 83% utilizaram efetivamente o serviço pela tecnologia Banda Larga Móvel. Fixando a taxa de penetração do acesso ao serviço de internet em 53,49%, apresentado um crescimento de 13,3% em relação a 2014. O acesso ao serviço de internet pela tecnologia ADSL contou com 15.188 assinaturas em Dezembro de 2015, apresentando assim um decréscimo de 13% em relação ao período homólogo.
- Quanto ao serviço de televisão por assinatura em Dezembro de 2015 o serviço contava com 11.457 assinantes, com um crescimento de 1,5% em relação ao período homólogo, e uma taxa de penetração de 2,18%.

No final de 2017 o número de assinantes do serviço de telefonia fixa foi de 64.970, correspondendo a uma taxa de penetração de 12 acessos por 100 habitantes.

Em Dezembro de 2017 constavam 642.014 cartões SIM ativos no mercado móvel em Cabo Verde, o que representa um ligeiro aumento de cerca de 1% em relação ao mesmo período de 2016. Fixando a taxa de penetração neste período em 119%.

No que concerne o serviço de acesso à internet, no quarto trimestre de 2017 registou-se 396.868 assinaturas, dos quais 89% utilizaram efetivamente o serviço

pela tecnologia Banda Larga Móvel small screen. Fixando a taxa de penetração do acesso ao serviço de internet em 73%, apresentado um crescimento de 15% em relação ao mesmo período em 2015. O acesso ao serviço de internet pela tecnologia ADSL contou com 14.493 assinaturas em Dezembro de 2017, apresentando assim um aumento de 1% em relação ao período homólogo do ano passado.

Quanto ao serviço de televisão por assinatura em Dezembro de 2017 o serviço contava com 9.819 assinantes, com um decréscimo de 9% em relação ao período homólogo do ano passado, e uma taxa de penetração de 2%.

1.2. Motivação

Na atualidade, pessoas procuram formas de realizar as suas tarefas diárias de forma a gerirem melhor o seu tempo e a diminuir cada vez mais o seu esforço, mas, sempre tendo em vista a eficiência e eficácia, num mundo em que se vive em função de um programa pré-definido ou agenda. Sendo um fator motivador criar algo que ajude nisso.

Tendo em vista a gestão do tempo, a procura das farmácias às vezes tem-se tornado uma dor de cabeça para os mais apertados na agenda ou para as pessoas de pavor curto. Às vezes, encontrar uma farmácia aberta no momento pode não ser algo fácil, ou, ao encontrar há que está aberta pode-se ter que esperar para ser atendido, e, quando chega a sua vez o problema passa agora em encontrar o produto disponível.

Outro fator motivador, é que com as tecnologias se avançando as pessoas procuram muito a utilização de aplicativos nas suas tarefas diárias entre outras, então, queria trazer mais uma app que de alguma forma ajudaria as pessoas na realização delas.

Assim, vi a possibilidade de elaborar algo inovador que pudesse solucionar os problemas do dia-a-dia dos usuários criando uma aplicação que servisse de intermediação entre a população e as farmácias, onde tive todo o apoio do meu orientador.

1.3. Objetivos

1.3.1. Objetivos Gerais

O objetivo geral é desenvolver um protótipo de um aplicativo para *Smartphone* com sistema operacional Android que permita ao utilizador encontrar um produto numa farmácia aberta no momento, entre outras funcionalidades.

1.3.2. Objetivos Específicos

Os objetivos específicos do trabalho são:

- Implementar o sistema utilizando a plataforma Android;
- Utilizar banco de dados *MySQL*;
- Conectar a base de dados com o Android utilizando php;
- Utilizar o apoio a uma plataforma Web para a manutenção da base de dados;
- Saber qual farmácia esta aberta e com o produto desejado;
- Mostrar a rota para as farmácias;
- Acompanhar o atendimento das senhas no seu Smartphone;
- Ver informações das farmácias;
- O farmacêutico pode iniciar sessão pelo Smartphone na sua base de dados e manipular a mesma, adicionar, listar, atualizar e eliminar produtos da base de dados;
- Enviar email tanto de utentes como de farmacêuticos reportando erros ou melhorias.

1.4. Metodologia

Primeiro utilizou-se a internet para a parte teórica e recolha das referências bibliografias, depois, foi feito um levantamento dos requisitos do sistema, a modelagem do Sistema, o estudo da plataforma Android, Web Service, plataformas web e foram instalados e configurados os ambientes de desenvolvimento.

Também, deu-se a elaboração do protótipo seguido da implementação do sistema e foram feitos testes para avaliar as funcionalidades do sistema para saber se está tudo bem ou precisa de melhorias.

1.5. Estrutura do Trabalho

O trabalho é estruturado em seis capítulos, onde primeiro podemos ver a contextualização, a motivação, objetivo geral, objetivos específicos e a metodologia.

Depois, no segundo uma abordagem da computação móvel e da plataforma Android. No terceiro capítulo é sobre a arquitetura do sistema, análise dos requisitos e a modelagem do sistema. No quarto capítulo é onde podemos ver as ferramentas utilizadas no trabalho e a tecnologia utilizadas no desenvolvimento do projeto. Na quinta fase a demonstração de trechos de código e de telas do protótipo. No sexto e último capítulo, encontram-se as considerações finais e a conclusão do trabalho.

2. Enquadramento Teórico

Onde serão apresentados conceitos sobre a computação móvel e a plataforma Android.

2.1. Computação Móvel

A tecnologia caminha a passos largos rumo a um futuro promissor, a cada dia vemos avanços significativos em hardware e softwares que cada vez mais facilitam nossa vida. Hoje podemos ir a qualquer loja de informática e adquirir aparelho que a 10 anos era impossível prever por muitos da área como iPhone e iPad.

Segundo Weiser no futuro os computadores estariam tão entrelaçados em nosso cotidiano que se tornaram indistinguíveis criando assim o conceito da Computação Ubíqua.

Esse conceito é particularmente incrível, é de forma sorrateira de certa forma já estamos vivenciando um pouco disso em nosso dia a dia. Por exemplo, quando estamos utilizando acesso a internet enquanto estamos em um ônibus a caminho da faculdade é uma forma de computação ubíqua, pois não estamos visualizando os equipamento ou computadores que possibilitam esse serviço mas ele está ali disponível no ambiente de forma sutil. Através de um Smartphone acessamos diversos serviços como rede social, portais acadêmicos, assistimos filmes e lemos as últimas notícias através da rede e computadores que não temos a menor noção de onde esteja.

Segundo Weiser o conceito computação ubíqua e a união de dois outros dois conceitos importantes: computação móvel que é muito frequente no exemplo que mencionado acima e a computação pervasiva.

2.1.1. Dispositivos para Computação Móvel

Esse conceito apresenta a ideia de disponibilização e recursos computacionais onde o usuário esteja, ou seja móvel. Em termos simples é a facilidade de utilizar um dispositivos móvel como celular ou notebook para acessar a internet, enviar e-mails e ler notícias e até mesmo trabalhar mesmo estando em movimento como dentro de um ônibus indo do caminho de casa para o trabalho.

A grande limitação da computação móvel é que os dispositivos não são capazes de ajustar as informações conforme o contexto que esteja inserido e ajusta-las automática, ou seja ela apenas estende os serviço que você já possui em um micro computador em casa para um dispositivo que permite utiliza-los em movimento. Segundo Regina Araújo, uma solução para acomodar as mudança de ambiente dos dispositivos seria os usuários controlarem e configurar a aplicação a medida em que se movam o que seria inviável e inaceitável pela maioria dos usuários.

A uma grande necessidade de desenvolvimento de novos meios de integração entre o usuários e os computadores de forma mais precisas com o mínimo de erros. Hoje temos tecnologias de comunicação potentes capazes de disponibilizar serviços em praticamente qualquer lugar através da internet. A indústria a cada disponibiliza novas tecnologias que permitem a implementação de novos recursos móvel e pervasiva.

Os televisores inteligentes controlados por gestos e comando de voz já são uma realidade hoje, os smartphones e notebooks e as tvs inteligentes já estão interagindo entre si. Tecnologias como o Kinect da Microsoft permite a integração por voz e gestos como nenhum outro dispositivo permitiu antes. A possibilidade de criar residências inteligentes, estabelecimento inteligentes e veículos inteligentes já é possível com a tecnologia atual.

2.1.2. Tipos de Aplicativos Móveis

Podemos usar diferentes tipos de aplicativos para proteger nosso dispositivo.

2.1.2.1. Aplicativo Nativo

Os aplicativos nativos são desenvolvidos especificamente para uma plataforma de SO móvel que suporta linguagens de programação como Swift e Objective-C para iOS e Java™ para Android. O aplicativo é desenvolvido pelas diretrizes da plataforma para o desenvolvimento do aplicativo e usa as APIs que o SO oferece. O aplicativo fornece fácil acesso aos recursos de dispositivos integrados, como acesso à câmera, Bluetooth e GPS. Os aplicativos nativos executam mais rapidamente, fornecem uma aparência consistente e oferecem mais segurança. No entanto, os aplicativos nativos demoram mais para se desenvolver e precisam ser desenvolvidos para cada plataforma.

2.1.2.2. Aplicativo Web

Os aplicativos webs recebem este nome porque são executados na internet. Ou seja, os dados ou os arquivos que você trabalha são processados e armazenados dentro da web. Estes aplicativos geralmente não precisam ser instalados no seu computador.

O conceito de aplicativos webs está relacionado com o armazenamento na nuvem. Toda a informação é guardada de forma permanente em grandes servidores de internet e estes enviam a nossos dispositivos ou computadores os dados requeridos neste momento, deixando uma cópia temporária dentro deles.

Estes grandes servidores de internet que prestam o serviço de alojamento estão localizados ao redor do mundo, e assim fazem com que o serviço prestado não seja assim tão caro e na maioria dos casos gratuitos e extremamente seguros.

Serviços como Gmail e Yahoo! São executados nos navegadores e executam as mesmas tarefas que programas de correio eletrônico como Microsoft Outlook.

Depois de se registrar a um serviço de correio eletrônico, você pode utilizá-lo imediatamente, sem requerimentos de instalação. Os seus e-mails permanecem armazenados na nuvem.

2.1.2.3. Aplicativo Híbrido

Aplicativos híbridos são aplicativos da web que são desenvolvidos com HTML5 e JavaScript. Esses aplicativos são websites que são empacotados para se comportar como um aplicativo nativo. Os aplicativos híbridos são executados em iOS e Android, e todos os dados do aplicativo são fornecidos por servidores de aplicativos. Esses aplicativos fazem chamadas de API de Serviço da Web para buscar os dados. Os aplicativos híbridos usam plataformas de desenvolvimento de aplicativo, como IBM® Worklight, Cordova, Angular ou Ionic. As plataformas de desenvolvimento de aplicativo fornecem o wrapper nativo para esses aplicativos. Os aplicativos híbridos executam mais lentamente porque os dados são carregados pela Internet por servidores de aplicativos.

2.1.3. Principais Plataformas de Desenvolvimento

O desenvolvimento de aplicações para mobile possui particularidades do desenvolvimento tradicional devido as limitações tanto do processamento, tamanho de tela e área de trabalho, além de estar sempre bombardeado por configurações distintas tanto do hardware quanto do software por parte dos fabricantes e suas concorrências.

Os principais sistemas operacionais do mercado atual são o Android, iOS, BlackBerry, HP webOS, Symbian OS, Bada da Samsung, e Windows Mobile que suportam aplicações binárias típicas de PC's com códigos de execução nativas da máquina. Windows Mobile pode ser compilado até em um x86 sem a necessidade de um emulador, e também podem suportar aplicações portáteis desenvolvidas com as tecnologias .NET Framework, Windows Mobile, Android, HP webOS e iOS (Apple) através de IDE de distribuição livre.

2.2. Android

Android é uma plataforma desenvolvida pelo Google voltada para dispositivos móveis totalmente aberta e livre (Open Source), que foi divulgada em 5 de novembro de 2007. Inicialmente o sistema Android foi desenvolvido pelo Google mas atualmente essa plataforma é mantida pela OHA (Open Handset Alliance), um grupo constituído por aproximadamente 84 empresas, as quais se uniram para inovar e acelerar o desenvolvimento de aplicações e serviços, com o objetivo de trazer aos consumidores uma experiência mais rica em termos de recursos e menos dispendiosa no mercado móvel.

2.3. IOS

IOS é o sistema da Apple para os seus dispositivos móveis. A versão mais recente do sistema operacional que equipa smartphones, tablets e setboxes desenvolvidos pela Apple é o iOS 7.1. Disponível para dispositivos como iPhone, iPod Touch, iPad e Apple TV, o iOS tem como destaque a integração com outros sistemas da Apple, como o Mac OS X.

O sistema, desenvolvido para dispositivos com suporte a telas touchscreen capacitivas, foi o primeiro a introduzir gestos como “pinça”, para ampliar ou diminuir a exibição de conteúdo na tela. A principal característica da interação com o iOS é o uso dos dedos para gerir a interface, dispensando canetas stylus.

2.4. Windows Phone

O Windows Phone é uma versão móvel do Windows. Desenvolvido pela Microsoft, a plataforma foi lançada em 2010 como sucessora do Windows Mobile, de 2000, que não fez sucesso tão grande como o esperado. Atualmente, ele está na versão 8, mas com a 8.1 liberada em edição beta via download nos próprios celulares.

A grande expoente é a Nokia, que, inclusive, foi comprada pela Microsoft neste ano, e a série de aparelhos Lumia é a principal demonstração do sistema que ainda fica

devido um pouco em relação ao Android e ao iOS. Isso, porém, é compreensível, pois chegou ao mercado quando os rivais já estavam consolidados.

2.5. O Android

O Android é uma pilha de software com base em Linux de código aberto criada para diversos dispositivos e fatores de forma. O diagrama a seguir mostra a maioria dos componentes da plataforma Android. A fundação da plataforma Android é o kernel do linux. O Android Runtime (ART) confia no kernel do Linux para cobrir funcionalidades como encadeamento e gerenciamento de memória de baixo nível.

2.5.1. Arquitetura do Android

A figura abaixo pode-se verificar a arquitetura do Android e as suas diversas camadas e componentes.

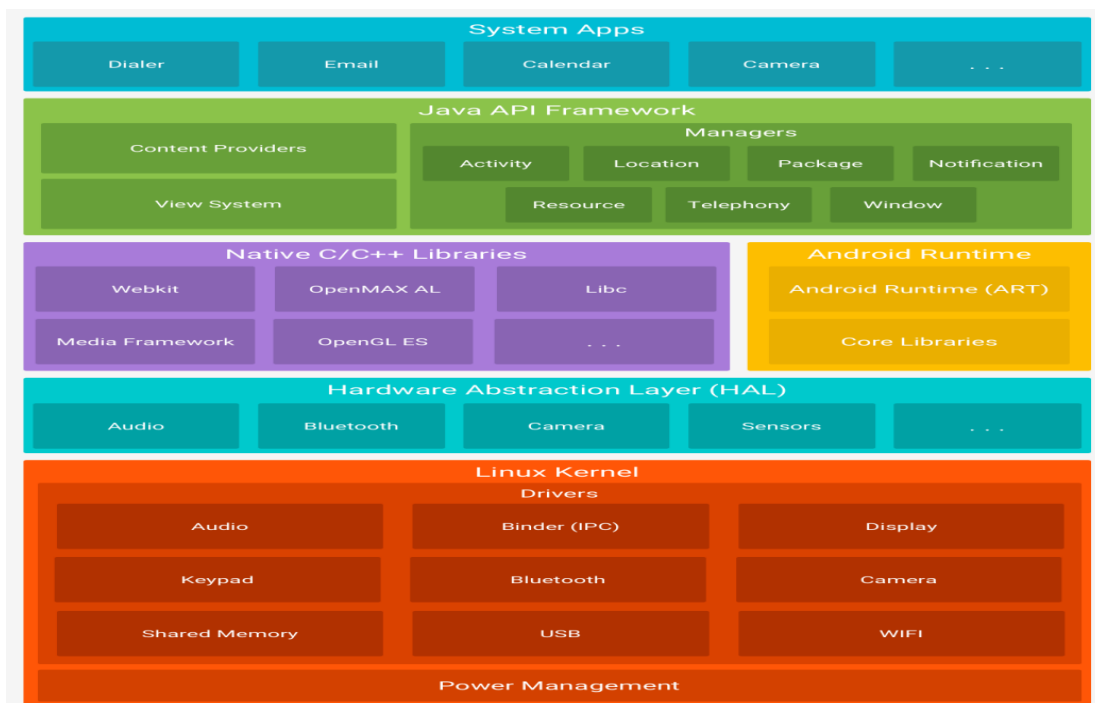


Figura 1: Arquitetura plataforma Android

Fonte: <http://www.softblue.com.br/blog/home/postid/11/CONHECENDO+O+ANDROID>

2.5.2. Applications

Temos a camada de aplicações e as funções básicas do dispositivo. Esta é a camada de interação entre o usuário e o dispositivo móvel, nela encontramos aplicativos cliente de e-mail, programa de SMS, calendário, mapas, navegador, contatos entre outros.

2.5.3. Applications Framework

Temos a camada de framework de aplicação (Application Framework), programas que gerenciam as aplicações básicas do telefone. Os desenvolvedores têm acesso total ao framework como um conjunto de ferramentas básicas com o qual poderá construir ferramentas mais complexas.

2.5.4. Bibliotecas e Android Runtime

Temos as camadas de bibliotecas (Libraries) e tempo de execução (Android RunTime).

A camada de biblioteca é um conjunto de instruções que dizem ao dispositivo como lidar com diferentes tipos de dados, incluindo um conjunto de biblioteca C / C++ usadas por diversos componentes do sistema e são expostas a desenvolvedores através da estrutura de aplicativo Android.

A camada de tempo de execução inclui um conjunto de bibliotecas do núcleo Java (Core Libraries). Para desenvolver aplicações para o Android, os programadores utiliza a linguagem de programação Java, nesta camada encontraremos a Máquina Virtual Dalvik (DVM).

O Android usa as máquinas virtuais Dalvik para rodar cada aplicação com seu próprio processo. Isso é importante por algumas razões: nenhuma aplicação é dependente de outra e se uma aplicação parar, ela não afeta quaisquer outras aplicações rodando no dispositivo e isso simplifica o gerenciamento de memória,

pois a máquina virtual está baseada em registradores e desenvolvida de forma otimizada para requerer pouca memória e permitir que múltiplas instâncias executem ao mesmo tempo.

Ao contrário do que se afirma, que a Dalvik é uma máquina virtual Java, isso não é verdadeiro, pois ela executa seu próprio tipo de bytecodes.

2.5.5. Linux Kernel

Temos a base da pilha, ou seja, o Kernel (Linux Kernel), para desenvolvê-la foi utilizado à versão 2.6 do Sistema Operacional Linux. Nele encontraremos os programas de gerenciamento de memória, configurações de segurança e vários drivers de hardware.

2.6. Componentes de uma Aplicação

A estrutura de aplicativo do Android permite criar aplicativos avançados e inovadores usando um conjunto de componentes reutilizáveis. Esta seção explica como criar os componentes que definem os blocos de construção do aplicativo e como conectá-los usando intents.



Figura 2: Componente de um aplicativo Android

Fonte: <https://elandroidelibre.espanol.com/2014/02/aprende-android-en-20-conceptos-conceptos-1-y-2.html>

2.6.1. Activities

É um componente de aplicativo que fornece uma tela com a qual os usuários podem interagir para fazer algo, como discar um número no telefone, tirar uma foto, enviar um e-mail ou ver um mapa. Cada atividade recebe uma janela que exibe a interface do usuário. Geralmente, a janela preenche a tela, mas pode ser menor que a tela e flutuar sobre outras janelas.

2.6.2. Services

É um componente do aplicativo que pode realizar operações longas e não fornece uma interface do usuário. Outro componente do aplicativo pode iniciar um serviço e ele continuará em execução em segundo plano mesmo que o usuário alterne para outro aplicativo. Além disso, um componente poderá se vincular a um serviço para interagir com ele e até estabelecer comunicação entre processos (IPC). Por exemplo, um serviço pode lidar com transações de rede, reproduzir música, executar E/S de arquivos, ou interagir com um provedor de conteúdo, tudo a partir do segundo plano.

2.6.3. Content Providers

O provedor de conteúdo gerencia o acesso a um repositório central de dados. Um provedor é parte de um aplicativo do Android, que, em geral, fornece a própria IU para trabalhar com os dados. Contudo, provedores de conteúdo se destinam principalmente ao uso por outros aplicativos, que têm acesso ao provedor usando um objeto cliente do provedor. Juntos, provedores e clientes do provedor oferecem interface padronizada e consistente para dados que também lidam com comunicação em processos internos e garantem acesso a dados.

2.6.4. Broadcast Receiver

Recetores de transmissão são componentes que respondem a anúncios de transmissão por todo o sistema. Muitas transmissões se originam do sistema — por exemplo, uma transmissão que anuncia que uma tela foi desligada, a bateria está

baixa ou uma tela foi capturada. Os aplicativos também podem iniciar transmissões — por exemplo, para comunicar a outros dispositivos que alguns dados foram baixados no dispositivo e estão disponíveis para uso. Embora os recetores de transmissão não exibam nenhuma interface do usuário, eles podem criar uma notificação na barra de status para alertar ao usuário quando ocorre uma transmissão. Mais comumente, no entanto, um recetor de transmissão é somente um "portal" para outros componentes e realiza uma quantidade mínima de trabalho. Por exemplo, ele pode iniciar um serviço para executar um trabalho baseado no evento.

Os recetores de transmissão são implementados como subclasses de Broadcast Receiver e cada transmissão é entregue como um objeto Intent. Para obter mais informações, consulte a classe Broadcast Receiver.

2.6.5. Android Manifest

Antes de o sistema Android iniciar um componente de aplicativo, é preciso ler o arquivo `AndroidManifest.xml` do aplicativo para que o sistema saiba que o componente existe. O aplicativo precisa declarar todos os seus componentes nesse arquivo, que deve estar na raiz do diretório do projeto do aplicativo.

O manifesto para além de declarar os componentes do aplicativo ele também identifica todas as permissões do usuário de que o aplicativo precisa, como acesso à internet ou acesso somente leitura aos contatos do usuário, declara o nível de API mínimo exigido pelo aplicativo com base nas APIs que o aplicativo usa, declara os recursos de hardware e software usados ou exigidos pelo aplicativo, como câmara, serviços de Bluetooth ou tela multitoque, entre outros.

2.7. Ciclo de Vida de uma Aplicação

Atividades no sistema são gerenciadas como um activity stack, ou pilha de atividades em português. Quando uma atividade é iniciada, ela é colocada no topo da pilha e se torna a atividade corrente - a atividade anterior sempre permanece abaixo na pilha e não vai ser mostrada enquanto a atividade corrente não terminar.

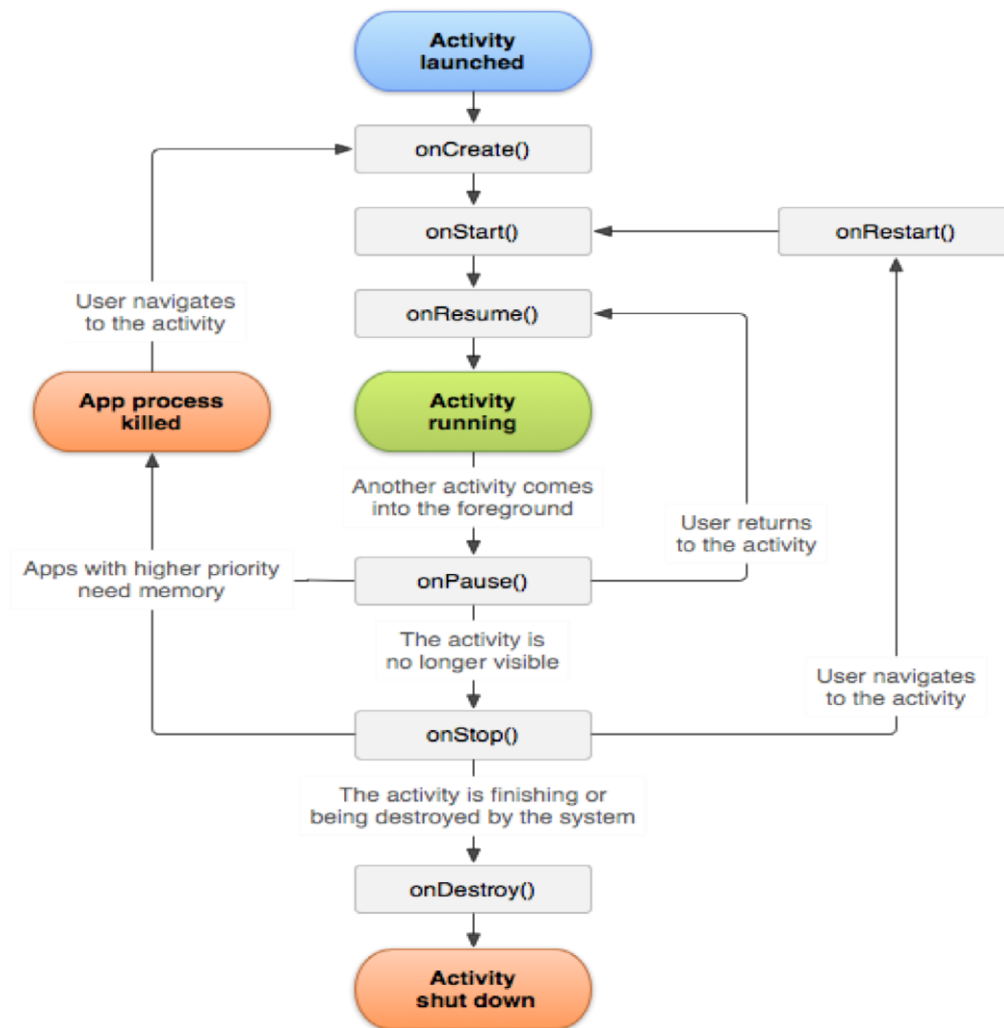


Figura 3: Ciclo de vida de um aplicativo Android

Fonte: <http://developer.android.com/guide/components/activities.html>

Uma *activity* é uma tarefa, muito focada, do que um usuário pode fazer. Quase todas as atividades interagem com o usuário, então uma classe de atividade toma conta da criação de uma janela para você onde você poderá colocar todos os componentes de UI (User Interface) com o `setContentView(view)`. Enquanto atividades são normalmente apresentadas para o usuário como telas full-screen, elas também podem ser apresentadas de outra maneira: como janelas flutuantes (através de um tema com `android:isFloating` configurado) ou embutido dentro de outra atividade (usando `ActivityGroup`). Existem dois métodos que quase todas as subclasses de `Activity` vão implementar:

onCreate() - É a primeira função a ser executada em uma Activity. Geralmente é a responsável por carregar os layouts XML e outras operações de inicialização. É executada apenas uma vez.

onStart() - É chamada imediatamente após a *onCreate()* – e também quando uma Activity que estava em background volta a ter foco.

onResume() - Assim como a *onStart()*, é chamada na inicialização da Activity e também quando uma Activity volta a ter foco. Qual a diferença entre as duas? A *onStart()* só é chamada quando a Activity não estava mais visível e volta a ter o foco, a *onResume()* é chamada nas “retomadas de foco”.

onPause() - É a primeira função a ser invocada quando a Activity perde o foco (isso ocorre quando uma nova Activity é iniciada).

onStop() - Só é chamada quando a Activity fica completamente encoberta por outra Activity.

onDestroy() - A última função a ser executada. Depois dela, a Activity é considerada “morta” – ou seja, não pode mais ser relançada. Se o usuário voltar a requisitar essa Activity, um novo objeto será contruído.

onRestart() - Chamada imediatamente antes da *onStart()*, quando uma Activity volta a ter o foco depois de estar em background.

3. Análise do Sistema a Desenvolver

Neste capítulo falarei sobre a análise dos requisitos do meu sistema e a sua modelagem feita através de anotações gráficas através de diagramas da UML.

3.1. Visão Geral do Sistema

Haverá um servidor Web alojando a base de dados MYSQL que conterá informações sobre as farmácias que podem ser acessadas através de códigos que foram feitos em PHP em formato JSON, que será tratada e mostrada na tela da aplicação adequadamente, mas há que ter acesso a internet.

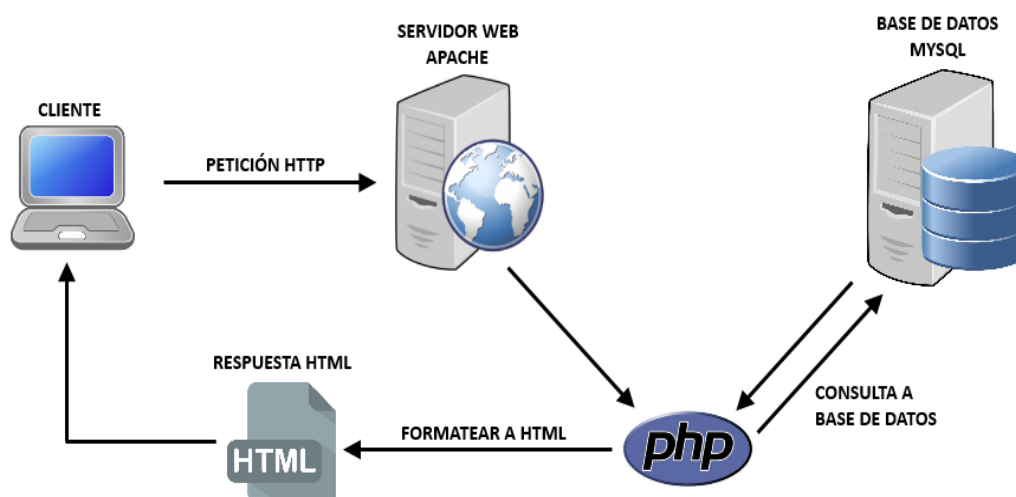


Figura 4: Arquitetura do sistema

Fonte: <http://mybrowserupdates.com/web-browser-web-server-and-web-client/>

3.2. Requisitos do Sistema

Os requisitos do sistema subdividem em funcionais e não funcionais:

Requisitos funcionais - define uma função de um sistema de software ou seu componente. O requisito funcional representa o quê o software faz, em termos de tarefas e serviços. Uma função é descrita como um conjunto de entradas, seu comportamento e as saídas. Os requisitos funcionais podem ser cálculos, detalhes

técnicos, manipulação de dados e de processamento e outras funcionalidades específicas que definem o que um sistema, idealmente, será capaz de realizar. Requisitos comportamentais, que descrevem todos os casos em que o sistema utiliza os requisitos funcionais, são extraídos dos casos de uso.

Requisitos não funcionais - são os requisitos relacionados ao uso da aplicação em termos de desempenho, usabilidade, confiabilidade, segurança, disponibilidade, manutenção e tecnologias envolvidas. Estes requisitos dizem respeito a como as funcionalidades serão entregues ao usuário do software.

3.2.1. Requisitos Funcionais da Aplicação

Mostra a farmácia aberta - mostrará a farmácia aberta num mapa permitindo ao utilizador visualizar a sua posição atual bem como a farmácia que se encontra mais próximo da pessoa com a informação do produto.

Acompanhar a fila - o sistema permitirá que o utilizador acompanhe com o seu Smartphone qual a senha a ser atendida no balcão.

Ver informações das farmácias - o sistema permitirá que o utilizador consiga ter acesso a informações das farmácias.

Rota das farmácias - o sistema permitirá que o utilizador veja a rota desde a localização do mesmo até a farmácia.

Realizar Login - o utilizador, cadastrado previamente pelo administrador, poderá fazer login através do seu nome de utilizador e da sua senha.

Listar produto - o sistema deverá permitir ao utilizador liste os produtos na base de dados.

Registrar produto - o sistema deverá permitir o registo de produtos, informando o nome, a categoria, o preço e a quantidade.

Atualizar produto - o sistema deverá permitir que o utilizador atualize um produto.

Eliminar produto - o sistema deverá permitir ao utilizador elimine produtos na base de dados.

Enviar email – o sistema permitirá que tanto o utilizador como o farmacêutico enviem emails reportando erros ou melhorias ao administrador.

3.2.2. Requisitos Não Funcionais

Neste tópico serão mostrados os requisitos não-funcionais do sistema desta pesquisa, agrupados pela sua classificação.

Usabilidade - A interface do sistema deverá se comportar adequadamente independente do front-end que será utilizado para acesso – Browser, Smartphone ou Tablet.

Operacional – é necessária uma plataforma Android 4.4 ou superior, obrigatório o uso de internet, deverá usar o GPS do dispositivo para fornecer informações sobre a posição do utilizador e a farmácia e mostrar a rota.

Implementação – utilizará o android SDK para a implementação do programa e para ter acesso a base de dados trechos de códigos desenvolvidos em PHP em formato JSON.

Arquitetura – será usado a plataforma Android, o MySQL para alojar a base de dados e gerenciamento da mesma, a comunicação é feita utilizando o PHP em formato JSON.

3.3. Modelagem do Sistema

Independente do porte de uma solução, todo projeto de software é caracterizado por um estágio inicial, sendo este momento uma fase de análise em que se procura estudar de que forma o mesmo será conduzido. A ênfase que é dada a este tipo de atividade depende, basicamente, não apenas do tamanho do sistema a ser construído, como também da forma como a equipe envolvida encontra-se estruturada e do conhecimento desta última acerca de padrões e metodologias próprios da área de software.

A UML contempla uma série de notações para a construção de diagramas representando diferentes aspectos de um software, além de não estar presa a metodologias ou tecnologias específicas de desenvolvimento. Sistemas construídos nas mais variadas linguagens e plataformas como C#, VB.NET, Java, Delphi etc. podem se beneficiar das vantagens decorrentes do uso desta linguagem.

A UML disponibiliza um conjunto de diferentes tipos de diagramas, dentre as quais serão mostrados:

- Diagrama de Use Cases;
- Diagrama de Classes;
- Diagrama de Sequência;
- Diagrama de Atividade;

3.3.1. Diagramas de Caso de Uso

Segundo Alberto da Silva & Carlos Videira (UML, Metodologias e Ferramentas CASE, 2001, Pag.145) um diagrama de use case ou caso de utilização é uma sequência de ações que um ou mais atores realizam num sistema de modo a obterem um resultado particular.

O modelo de casos de utilização permite capturar os requisitos de um sistema através do detalhe de todos os cenários que os utilizadores podem realizar. Os casos

de utilização, mais que iniciar a modelação de requisitos de um sistema, dirigem/conduzem todo o processo de desenvolvimento.

Podemos verificar esse diagrama na figura a seguir:

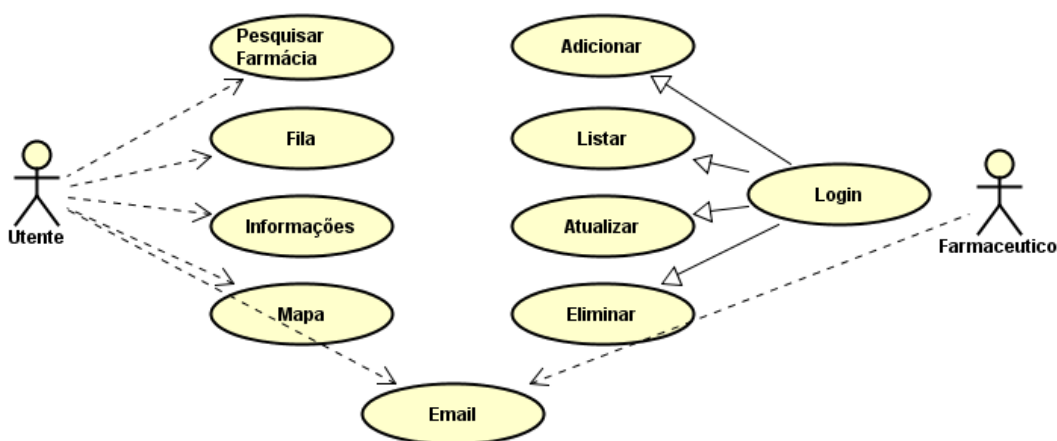


Figura 5: Diagrama de Caso de Uso

Fonte: Elaborado pelo autor

3.3.2. Diagrama de Classes

Segundo Alberto da Silva & Carlos Videira (UML, Metodologias e Ferramentas CASE, 2001, Pag.186) um diagrama de classes ilustra um conjunto de classes, interfaces, colaborações e respectivas relações, em geral de dependência, generalização e de associação.

3.3.3. Diagrama de Sequência

O diagrama de sequência ilustra uma interação segundo uma visão temporal. Um diagrama de sequência é representado através de duas dimensões: a dimensão horizontal, que representa o conjunto de objetos intervenientes; e a dimensão vertical que representa o tempo.

3.3.4. Diagrama de Atividade

Segundo Alberto da Silva & Carlos Videira (UML, Metodologias e Ferramentas CASE, 2001, Pag.222) um diagrama de atividades é um caso particular de um diagrama de estados, no qual todos ou a maioria dos estados são “estados de atividades” e todas ou a maioria das transições são desencadeadas pela conclusão das atividades dos estados anteriores.

Analise as figuras seguintes:

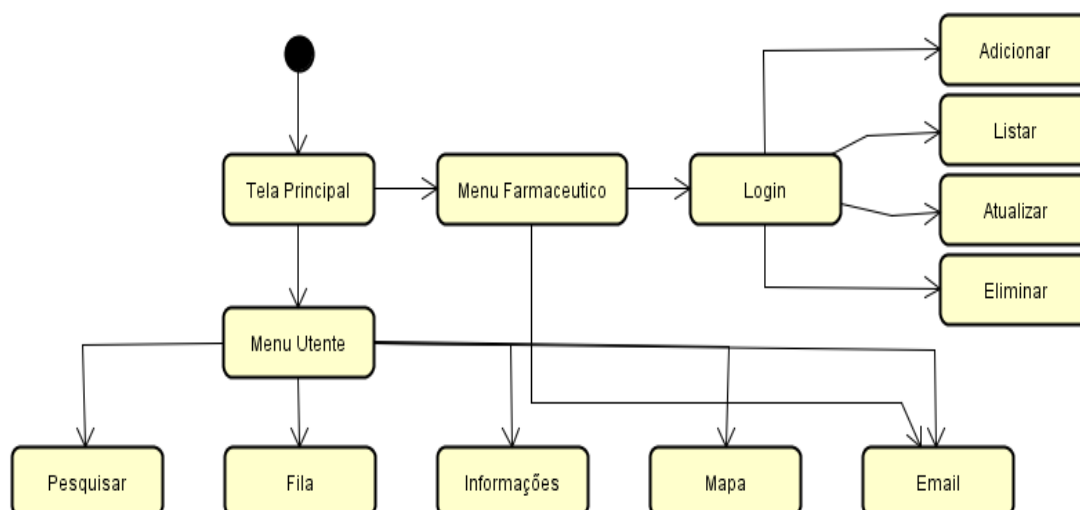


Figura 6: Diagrama de Atividade

Fonte: Elaborado pelo autor

3.3.5. Base de Dados

Foi utilizado o SGBD MySql para a criação de uma base de dados, armazenamento e gerenciamento. Segue a figura seguinte:

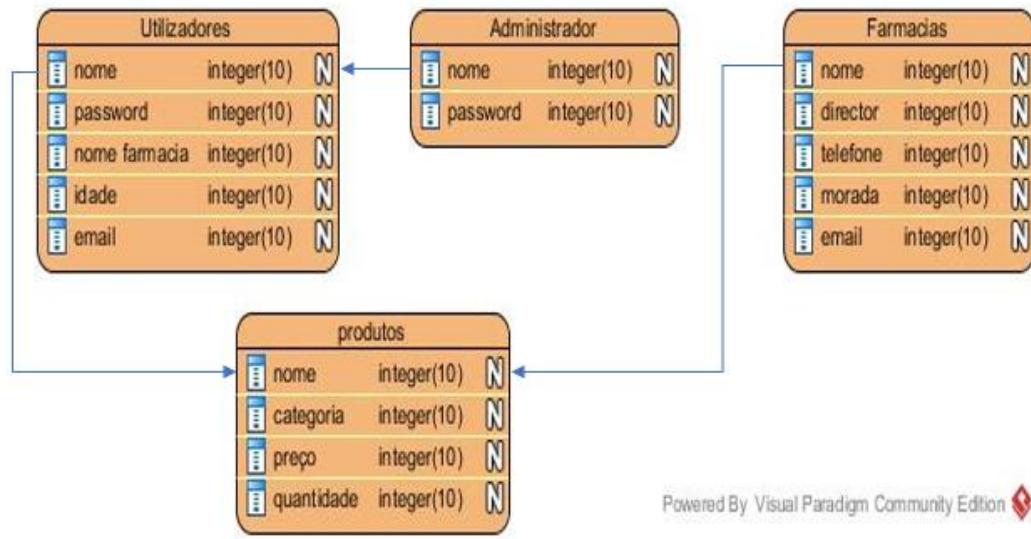


Figura 7: Base de dados

Fonte: Elaborado pelo autor

4. Ferramentas e Tecnologias Utilizadas

4.1. Android SDK

O Android SDK ou Kit de Desenvolvimento de Software para Android é um pacote com diversas ferramentas utilizadas pelo Android Studio e pelos desenvolvedores Android, incluindo componentes como o SDK Tools, Build Tools e o Platform Tools.

O Kit de Desenvolvimento de Software para Android permite aos desenvolvedores criarem aplicativos para a plataforma Android de forma nativa. O Android SDK inclui projetos de exemplo com código-fonte, ferramentas de desenvolvimento, emuladores e bibliotecas necessárias para criar os aplicativos Android.

Os aplicativos são escritos usando a linguagem de programação Java e executados na ART e Dalvik, máquinas virtuais personalizadas e projetadas para rodar dentro dos dispositivos Android que funcionam em cima de um kernel Linux.

4.2. Linguagem PHP

O PHP (um acrônimo recursivo para *PHP: Hypertext Preprocessor*) é uma linguagem de script open source de uso geral, muito utilizada, e especialmente adequada para o desenvolvimento web e que pode ser embutida dentro do HTML.

O que distingue o PHP de algo como o JavaScript no lado do cliente é que o código é executado no servidor, gerando o HTML que é então enviado para o navegador. O navegador recebe os resultados da execução desse script, mas não sabe qual era o código fonte. Você pode inclusive configurar seu servidor web para processar todos os seus arquivos HTML com o PHP, e então não há como os usuários dizerem o que você tem na sua manga.

A melhor coisa em usar o PHP é que ele é extremamente simples para um iniciante, mas oferece muitos recursos avançados para um programador profissional. Não tenha medo de ler a longa lista de recursos do PHP. Pode entrar com tudo, o mais rápido que puder, e começar a escrever scripts simples em poucas horas.

4.3. Base de Dados MySQL

O **MySQL** é um sistema gerenciador de banco de dados relacional de código aberto usado na maioria das aplicações gratuitas para gerir suas bases de dados. O serviço utiliza a linguagem SQL (Structure Query Language – Linguagem de Consulta Estruturada), que é a linguagem mais popular para inserir, acessar e gerenciar o conteúdo armazenado num banco de dados.

Para utilizar o MySQL, é necessário instalar um servidor e uma aplicação cliente. O servidor é o responsável por armazenar os dados, responder às requisições, controlar a consistência dos dados, bem como a execução de transações concomitantes entre outras. O cliente se comunica com o servidor através da SQL. A versão gratuita do MySQL é chamada de Edição da Comunidade e possui o servidor e uma interface gráfica cliente.

O servidor deve ser instalado e configurado para receber conexões dos clientes. No MySQL, o principal cliente é a interface gráfica cliente fornecida pela Oracle, denominada MySQL Workbench. Através do MySQL Workbench, pode-se executar consultas SQL, administrar o sistema e modelar, criar e manter a base de dados através de um ambiente integrado. O MySQL Workbench está disponível para Windows, Linux e Mac OS.

4.4. JSON

JSON (JavaScript Object Notation - Notação de Objetos JavaScript) é uma formatação leve de troca de dados. Para seres humanos, é fácil de ler e escrever. Para máquinas, é fácil de interpretar e gerar. Está baseado em um subconjunto da linguagem de programação JavaScript. JSON é em formato texto e completamente independente de linguagem, pois usa convenções que são familiares às linguagens C e familiares, incluindo C++, C#, Java, JavaScript, Perl, Python e muitas outras. Estas propriedades fazem com que JSON seja um formato ideal de troca de dados.

JSON está constituído em duas estruturas:

- Uma coleção de pares nome/valor. Em várias linguagens, isto é caracterizado como um *object*, record, struct, dicionário, hash table, keyed list, ou arrays associativas.
- Uma lista ordenada de valores. Na maioria das linguagens, isto é caracterizado como uma *array*, vetor, lista ou sequência.

4.5. Genymotion

Genymotion é um dos melhores emuladores Android para PC. Ele se distingue pela velocidade de funcionamento e permite verificar confortavelmente a operabilidade dos aplicativos. O emulador Genymotion é uma continuação avançada de AndroidVM. Ele engloba um círculo largo de utilizadores e pode ser usado em “tandem” com os sistemas operacionais Windows, MacOS e Linux.

Para o emulador funcionar corretamente é necessário instalar o VirtualBox no computador. Tem uma versão paga e gratuita dele. A primeira versão apresenta um espectro mais largo de possibilidades. Por exemplo, entre as suas qualidades vale a pena destacar a função Multi-touch. Ela será útil para estúdios grandes especializados no desenvolvimento de software para o sistema operacionais Android.

Particularidades de Genymotion:

- Rápido carregamento do emulador.
- Carregamento imediato das imagens Android que já foram definidas.
- Possibilidade de emulação da ligação Wi-Fi sem fios.
- Emulação da internet com auxílio de diferentes tecnologias.
- Disponibilidade do modo de visualização em tela cheia.
- Apoio ADB.
- Apoio de três OS: Windows, Mac e Linux.
- Possibilidade de carregar os aplicativos a partir do Google Play.
- Flash e cameras podem ser emulados.

Limitações do emulador Genymotion:

- Inexistência de interface na língua portuguesa.
- Não são emuladas chamadas de voz.

4.6. HTML

Quando entramos numa página web qualquer, normalmente estamos interessados em ver toda a informação ou o conteúdo dessa página. Essa informação pode estar na forma de texto, imagem ou vídeo por exemplo. Então esse conteúdo da página web é definido com a linguagem HTML (HyperText Markup Language).

Segundo a K19 (Desenvolvimentos Web com HTML, CSS E Javascript, 2014, Pag. 21) HTML é uma linguagem de marcação originalmente proposta por Tim Berners-Lee no final da década de 1980. O seu era criar um mecanismo simples que pudesse ser utilizado por qualquer pessoa que quisesse disseminar documentos científicos.

Desde sua proposta até os dias de hoje, a linguagem HTML sofreu diversas alterações. A cada versão, novos recursos são adicionados e problemas corrigidos. A versão mais atual da especificação da linguagem HTML é a 5.

4.6.1. Versão HTML5

Segundo Francisco Morais (HTML5 e CSS3 Domine a Web do futuro, 2012, Pag.27) o surgimento do HTML5 mudou muitas coisas no mundo do desenvolvimento web, com novos elementos, novas funcionalidades e diversas outras novidades que possibilitam experiências melhores e integrações que antes eram apenas desejos e sonhos dos desenvolvedores por aí. Mesmo com algumas funcionalidades ainda em processo de definição, já podemos aproveitar diversas das novidades que o HTML5 trouxe para o mundo. Algumas que são referentes ao código que escrevemos: novos elementos, mudanças de sintaxe e atributos personalizados.

4.7. Sistema de Posicionamento Global

O sistema de posicionamento global, mais conhecido pela sigla GPS (em inglês Global Positioning System) é um sistema de posicionamento por satélite que fornece a um aparelho recetor móvel a sua posição, assim como informação horária, sob quaisquer condições atmosféricas, a qualquer momento e em qualquer lugar na Terra, desde que o recetor se encontre no campo de visão de três satélites GPS (quatro ou mais para precisão maior). Encontram-se em funcionamento dois sistemas de posicionamento por satélite: o GPS americano e o GLONASS russo. Existem também dois outros sistemas em implementação: o Galileo da União Europeia e o Compass chinês. O sistema americano é detido pelo Governo dos Estados Unidos e operado através do Departamento de Defesa dos Estados Unidos.

Inicialmente o seu uso era exclusivamente militar, estando atualmente disponível para uso civil gratuito. No entanto, poucas garantias apontam para que em tempo de guerra o uso civil seja mantido, o que resultaria num sério risco para a navegação.

O GPS é utilizado na aviação geral e comercial e na navegação marítima, porém, atualmente é utilizado por diversas pessoas, que querem saber sua posição na sua própria cidade, e principalmente para viajar. Com um sistema de navegação por satélite integrando um GPS, é possível encontrar o caminho para um determinado local, saber a velocidade e direção do seu deslocamento. Atualmente, o sistema está sendo muito utilizado em automóveis, com um sistema de mapas.

Existem diversos tipos de GPS, de diversas marcas com soluções "tudo-em-um", como os externos que são ligados por cabo ou ainda por Bluetooth, e celulares modernos (Smartphone), que possuem o GPS integrado e acessível através de seus próprios aplicativos.

5. Implementação do Protótipo

Protótipo é o termo usado para se referir ao que foi criado pela primeira vez, servindo de modelo ou molde para futuras produções.

Segundo Robson Santos (2007), protótipos são modelos construídos para simular a aparência e a funcionalidade de um produto em desenvolvimento. Um protótipo é uma representação da interface com a qual o usuário pode interagir e oferece informações para propor mudanças e melhorias. Um protótipo pode ser:

- um pequeno guia para o usuário;
- uma simulação em papel;
- uma simulação usando o próprio computador a partir de uma ferramenta de prototipagem;
- uma versão inicial do programa;
- o sistema a ser substituído, em funcionamento.

5.1. Configuração da Aplicação

O AndroidManifest.xml, que se encontra na plataforma Android, encontra-se na pasta raiz do projeto, que serve como o “coração” do projeto, é obrigatório o seu uso porque nele ficam configurados todos os recursos utilizados pela aplicação, recursos como Activities, permissões, Services entre outros.

Na figura a seguir mostra o arquivo AndroidManifest.xml da aplicação Pharm Go juntamente com todas as suas configurações:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.user.pharmgo">

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission
android:name="android.permission.ACCESS_WIFI_STATE" />

    <uses-permission
android:name="android.permission.ACCESS_FINE_LOCATION" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <meta-data
            android:name="android.app.default_searchable"
            android:value=".PesquisarActivity" />

        <activity
            android:name=".PesquisarActivity"
            android:launchMode="singleTop">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <action android:name="android.intent.action.SEARCH" />
                <action android:name="android.intent.action.VIEW" />
            </intent-filter>

            <meta-data
                android:name="android.app.searchable"
                android:resource="@xml/searchable" />
        </activity>
        <activity android:name=".LoginActivity" />
        <activity android:name=".UserAreaActivity" />
        <activity android:name=".RegisterActivity" />
        <activity android:name=".FilaActivity" />
        <activity android:name=".ListarActivity" />
    </application>
</manifest>
```

```

    <activity android:name=".MailActivity" />
    <activity
        android:name=".MainActivity"
        android:label="@string/app_name"
        android:theme="@style/AppTheme.NoActionBar">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />

            <category
                android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".ShowAllProdutsActivity" />
        <activity android:name=".ShowSingleRecordActivity" />
        <activity android:name=".UpdateActivity" />
        <activity android:name=".InfoActivity" />
        <!--
            The API key for Google Maps-based APIs is defined as a
            string resource.
            (See the file "res/values/google_maps_api.xml").
            Note that the API key is linked to the encryption key
            used to sign the APK.
            You need a different API key for each encryption key,
            including the release key that is used to
            sign the APK for publishing.
            You can define the keys for the debug and release targets
            in src/debug/ and src/release/.
        -->
        <meta-data
            android:name="com.google.android.geo.API_KEY"
            android:value="@string/google_maps_key" />

        <activity
            android:name=".MapsActivity"
            android:label="@string/title_activity_maps"></activity>
    </application>
</manifest>

```

Figura 7: Android Manifest

Fonte: Elaborado pelo autor

5.2. Criando a Base de Dados Local

Utilizei o MySQL para criação da base de dados.

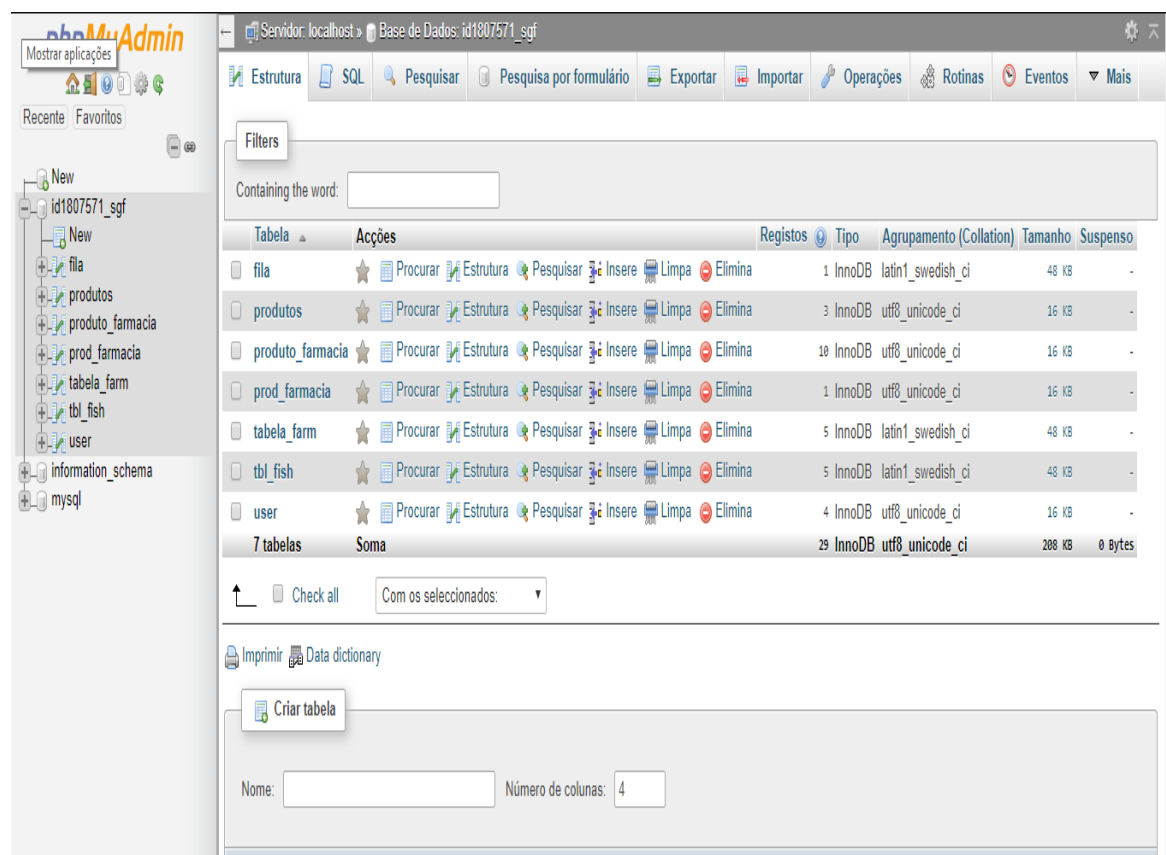


Figura 8: Base de dados Pharm Go

Fonte: Elaborado pelo autor

5.3. Pesquisando Produto

O utilizador introduz o produto procurado e a aplicação retorna com o mesmo, a farmácia aberta que tem o produto disponível e as opções de reserva ou compra, assim como mostra as figuras abaixo:

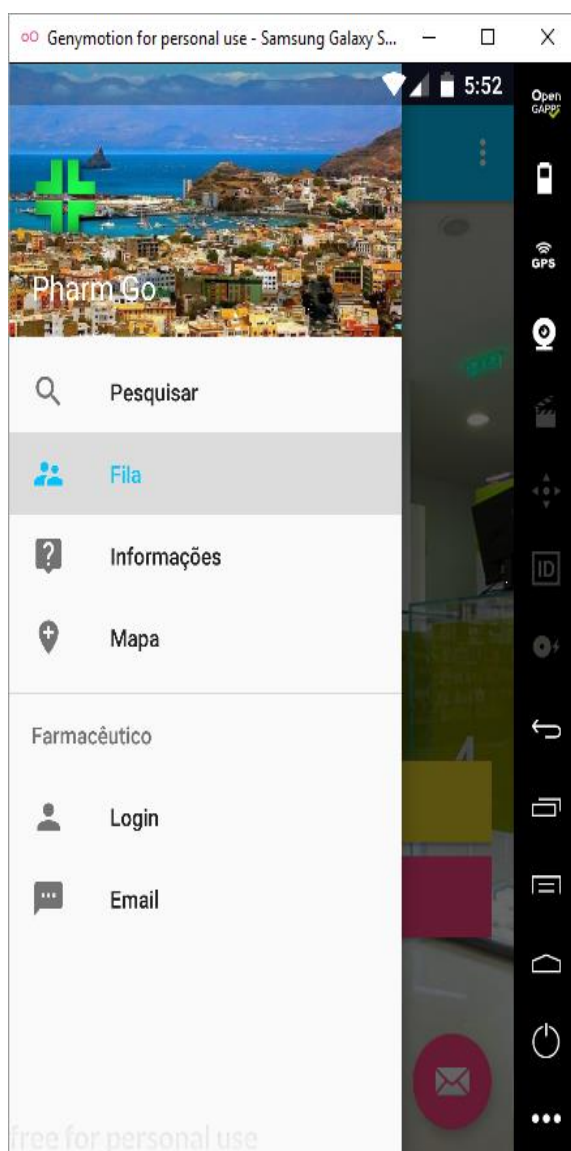


Figuras 9 e 10: Funcionalidade Pesquisar

Fonte: Elaborado pelo autor

5.4. Seguindo a Fila

O utilizador escolhe a farmácia que onde está ou para onde vai e pode acompanhar as senhas que estão sendo atendidas. Figuras abaixo:



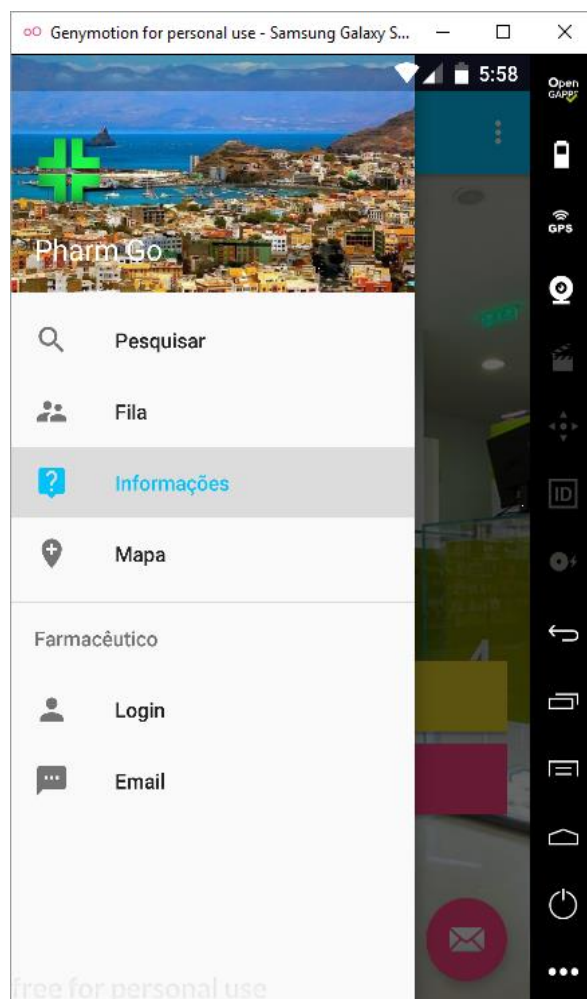
Figuras 11 e 12: Funcionalidade Fila

Fonte: Elaborado pelo autor

5.5. Ver Informações da Farmácia

O utilizador escolhe a farmácia que deseja e poderá ver informações da mesma.

Figuras abaixo:



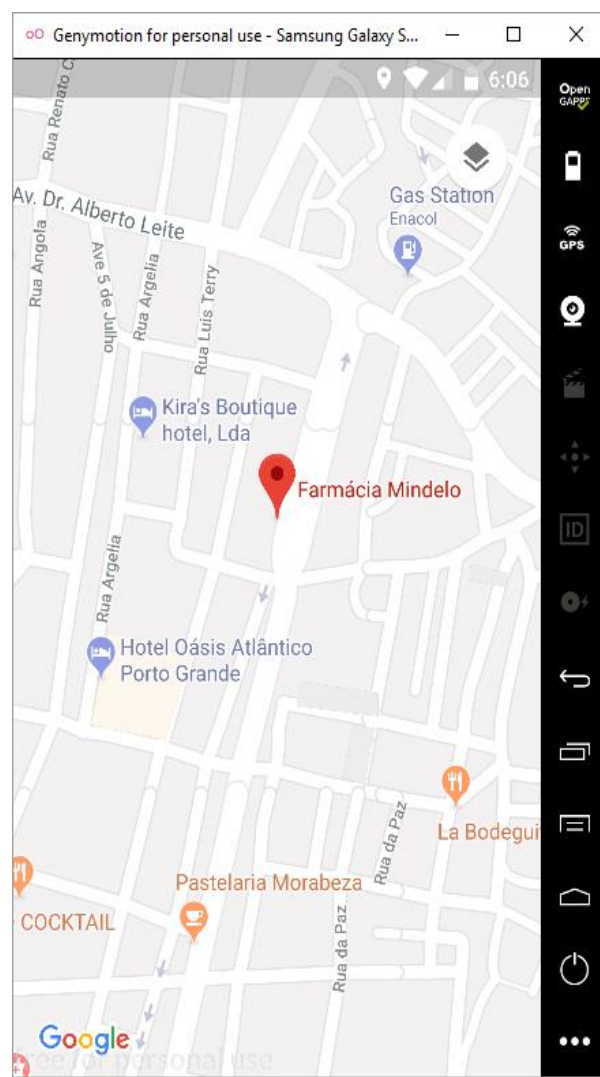
O utilizador pode fazer uma pergunta a farmácia, enviar alguma informação ou ainda poderá ver a sua base de dados.

Figuras 13 e 14: Funcionalidade Informações

Fonte: Elaborado pelo autor

5.6. Localização no Mapa

Se o utilizador não souber a localização de alguma das farmácias poderá ver a mesma na funcionalidade mapa. Apenas escolhe a farmácia desejada e a aplicação pega a localização da pessoa e faz uma rota até a farmácia.

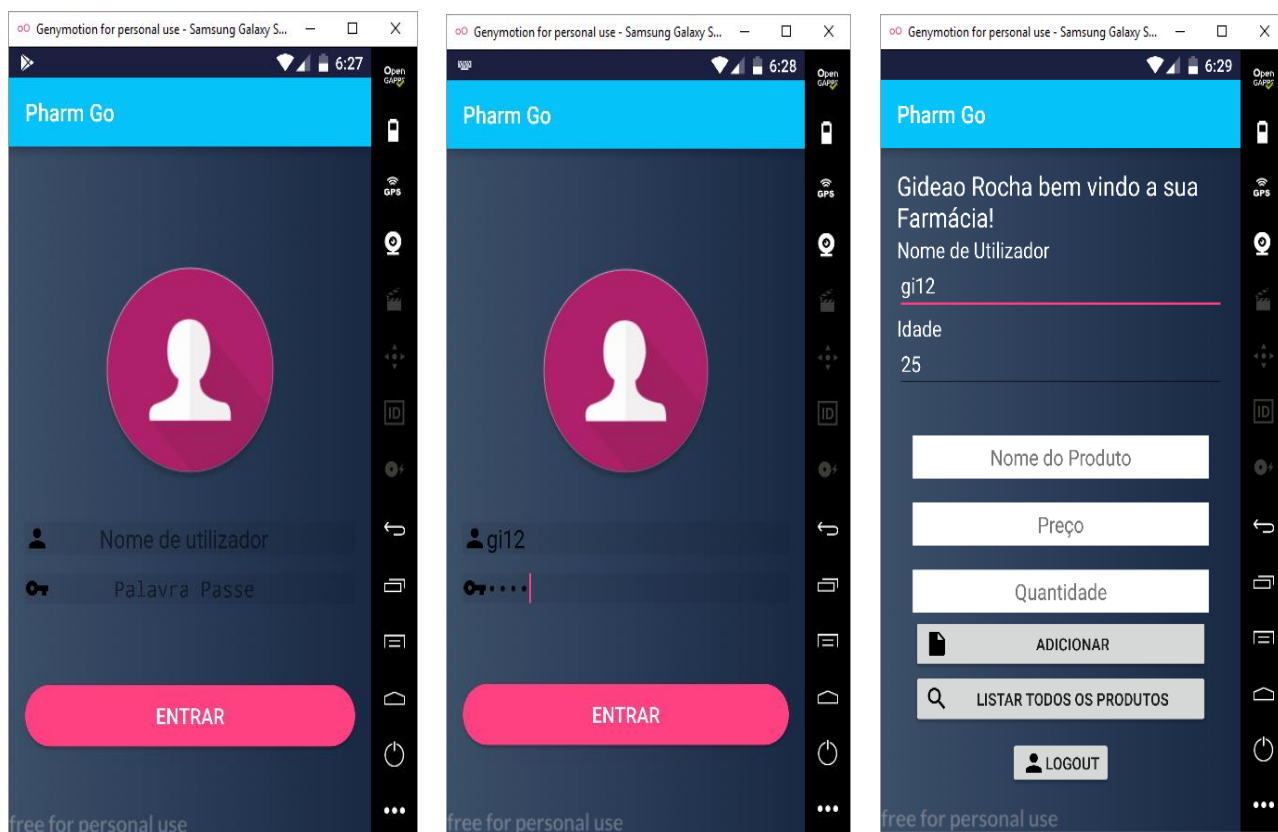


Figuras 15 e 16: Funcionalidade Mapa

Fonte: Elaborado pelo autor

5.7. Área Reservada apenas para Farmacêuticos

Apenas farmacêuticos podem fazer login por escolher a farmácia onde trabalham e introduzir o seu nome de utilizador e palavra-passe. Entrará numa área para utilizadores onde mostra o nome desse utilizador e sua idade, é uma forma de autenticação.

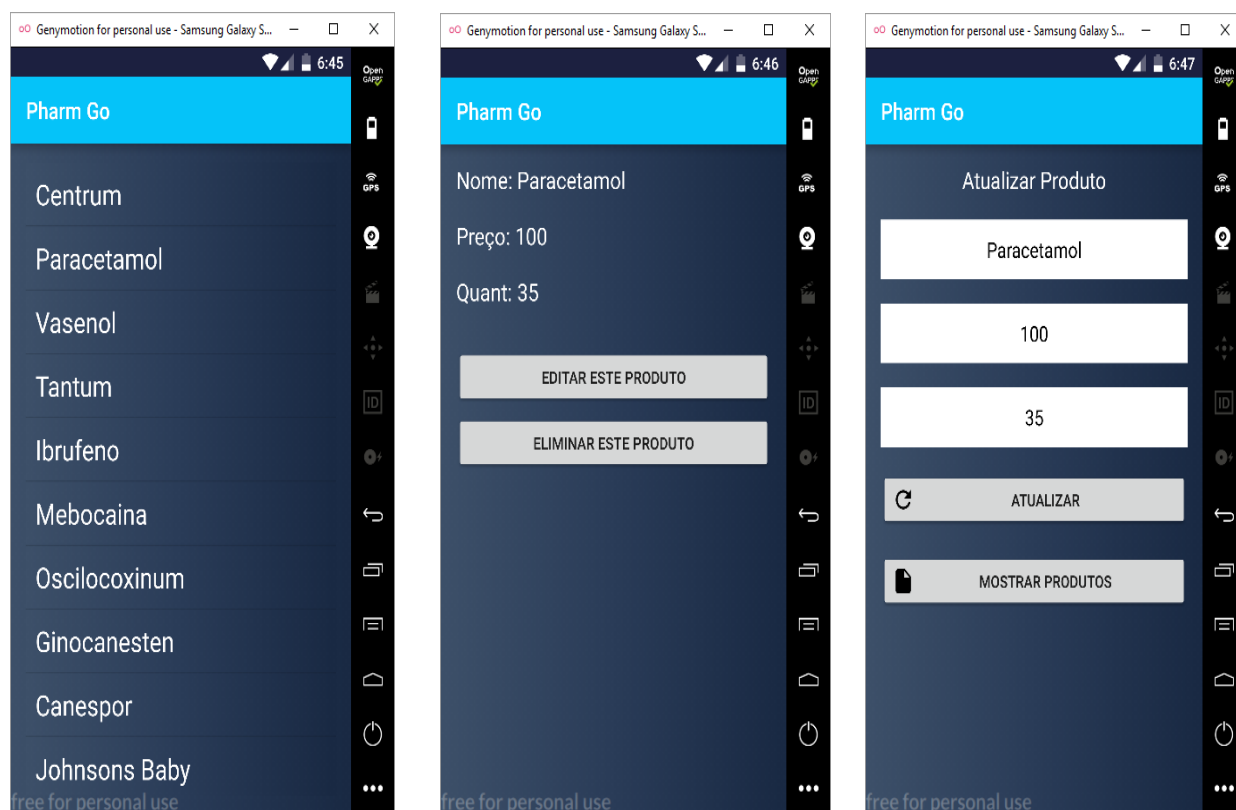


Figuras 17, 18 e 19: Funcionalidade Login, Adicionar e Listar

Fonte: Elaborado pelo autor

Logo ao entrar podemos ver 2 funcionalidades, o de adicionar um novo produto e o de listar todos os produtos.

Ao listar os produtos o farmacêutico poderá ver mais informações do mesmo, e, poderá ver as outras duas funcionalidades, atualizar e eliminar. Figuras abaixo:

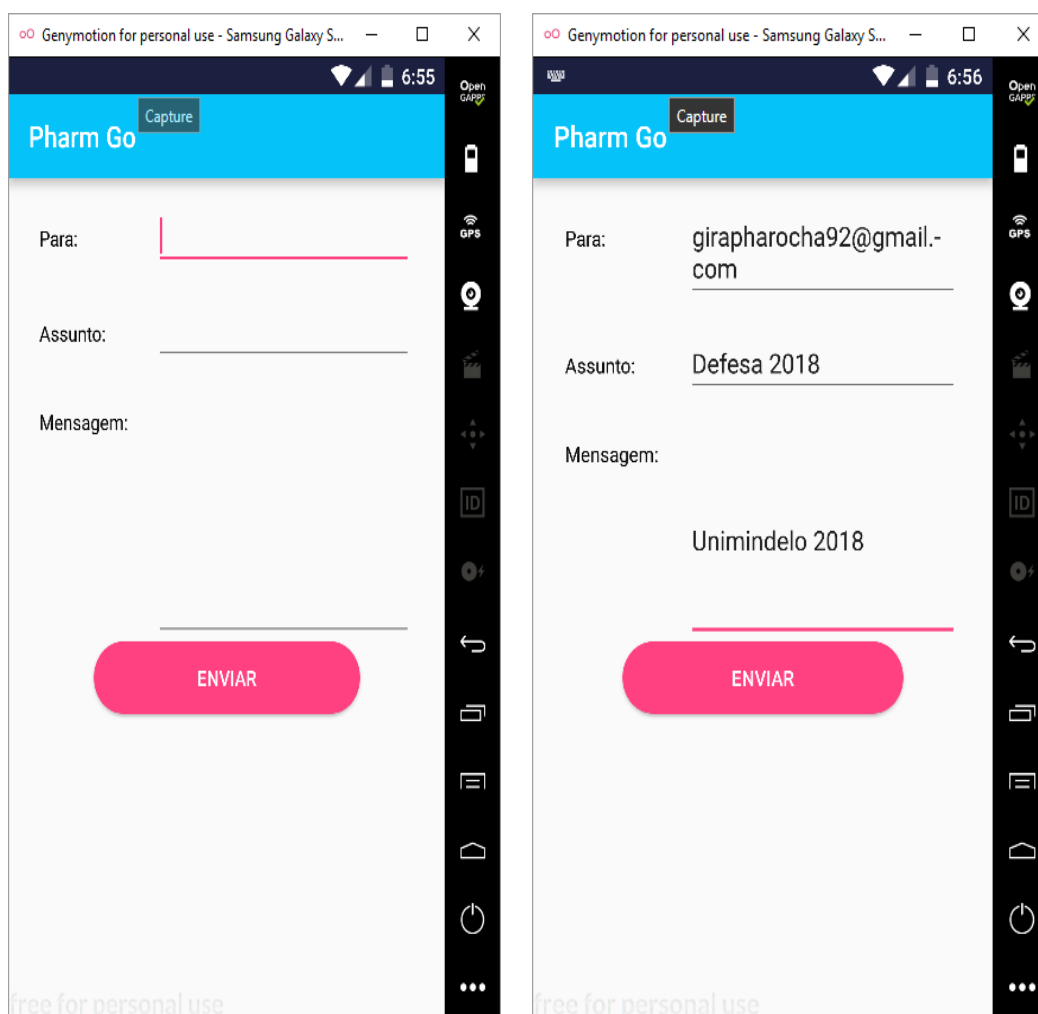


Figuras 20, 21 e 22: Funcionalidade Listar, Atualizar e Eliminar

Fonte: Elaborado pelo autor

5.8. Envio de Email

Tanto um utente como um farmacêutico podem enviar emails ao administrador reportando erros ou até mesmo melhorias. Figuras abaixo:



Figuras 23 e 24: Funcionalidade Email

Fonte: Elaborado pelo autor

5.9. Website

A página web é muito simples, tem as funcionalidades de login para farmacêuticos onde pode manipular os dados da sua farmácia, como adicionar, listar, atualizar e eliminar. Figuras abaixo:



Figuras 25: Website Login

Fonte: Elaborado pelo autor



Figuras 26: Página Inicial

Fonte: Elaborado pelo autor

6. Conclusão

6.1. Resultados Obtidos

Com este projeto tentou-se trazer algo de novo a população e ao mesmo tempo necessário, importante e vantajoso que tornará a busca pelas farmácias e os seus produtos uma tarefa mais fácil e eficiente.

A implementação desse projeto trará consigo várias vantagens também para quem se encontra numa fila de espera graças a ele um cliente ao chegar na farmácia e retirar a sua senha caso encontrar uma fila enorme, com muitas pessoas para serem atendidas na sua frente, já não terá problemas em sair do estabelecimento e ir para outro lugar resolver outro assunto. Com essa aplicação o cliente poderá controlar a sua vez de qualquer lugar desde que esse tenha acesso a internet. Poderá, também, ver as rotas até as farmácias e as informações das mesmas.

Com base nos resultados da etapa de avaliação dos aplicativos, verificou-se que este trabalho atingiu seu objetivo em desenvolver um aplicativo que provê aos seus utilizadores a facilidade e simplicidade das ferramentas desenvolvidas.

A Pharm Go dispõe das seguintes vantagens: gestão do tempo; facilidade de acesso as farmácias; evitar filas; facilidade de acesso a informações de produtos e farmácias; acompanhar a fila pelo Smartphone; gestão de stock; facilidade de acesso a base de dados e execução de manipulações a mesma; melhoria no atendimento aos utentes.

6.2. Trabalhos Futuros

Têm-se como perspectiva a expansão do aplicativo para a plataforma mobile nomeadamente a IOS e Windows Phone. Contudo, em caso de negócio poderá ser feito um ajuste na aplicação e dividi-lo em 3 partes:

- 1) Primeira parte – Apenas com a opção de pesquisar farmácias, onde posso também, adicionar outras funcionalidades práticas.
- 2) Segunda parte – com a opção de filas, onde todas as empresas caboverdianas podem ser cadastrados e disponibilizados os seus painéis de senhas no mobile.
- 3) Terceira parte – Pegar da parte dos farmacêuticos e coloca-los numa plataforma reservada apenas a eles.

7. Referências Bibliográficas

WEISER, M. “The Computer for the 21 Century”. *Scientific American*, Vol. 265, no.3, Setembro, 1991. Pag. 94-104.

ARAUJO, Regina Borges. *Computação Ubíqua: Princípios, Tecnologias e Desafios.* Universidade Federal de São Carlos- UFSCar, São Carlos – SP.

Landis, Joshua & Haff, G Gregory & Weatherford, Zach (2006), *Android Programming Guide for Beginners.*

Nixon, Robin (2014) *Learning PHP, MySQL, JavaScript, CSS & HTML5*, Third Edition.

Silva, Carlos, *Introdução a Programação Android.*

Lecheta, Ricardo R (2016) *Google Android, Aprenda a criar aplicativos para dispositivos móveis com SDK*, 5ª Edição.

Lopes, Sérgio (2013). *A Web Mobile, Programe para um mundo de muitos dispositivos.*

Mazza, Luiz (2012), *HTML5 e CSS3, Domine a web do futuro.*

Monteiro, João B., *Google Android, Crie aplicações para smarphone e tablets.*

IBM Knowledge Center - disponível em [Consultado em 11/12/2017 às 19:00]:

https://www.ibm.com/support/knowledgecenter/pt-br/SS8H2S/com.ibm.mc.doc/dev_source/references/dev_about_app_types.htm

Ciclo de Vida de uma aplicação android [Consultado em 11/12/2017 às 20:20]:

Disponível em <https://www.devmedia.com.br/entendendo-o-ciclo-de-vida-de-uma-aplicacao-android/22922>

DOMINGUES, Fabiano L., Computação Ubíqua, 2008 - disponível em [Consultado em 11/12/2017 às 21:00]:

www.hardware.com.br+artigos+computacao-ubiqua.html

Luis Felipe Zeni webservice [Consultado em 01/12/2017 às 10:40]:

Disponível em <https://www.youtube.com/user/morcegosx>

Alojamento dos ficheiros php [Consultado em 23/11/2016 às 20:20]:

Disponível em <https://www.000webhost.com/members/website/list>

Receber dados via json em um aplicativo android [Consultado em 13/12/2017 às 09:00]:

Disponível em <https://androidjson.com/android-json-parsing-listview-php-mysql/>

Tutorials Loop Android MySQL ListView [Consultado em 14/12/2017 às 10:55]:

Disponível em <http://camposha.info/source/android-php-mysql-ep-02-listview-select-show-httpurlconnection-source/>

Tutorials point. [Consultado em 15/12/2017 às 09:15]:

Disponível em https://www.tutorialspoint.com/android/android_studio.htm

Android Tutorial [Consultado em 16/12/2017 às 08:50]

Disponível em <http://androidcss.com/android/android-search-view-php-mysql/>

Introdução ao JSON [Consultado em 16/12/2017 às 16:00]

Disponível em <http://www.json.org/json-pt.html/>

Android Developers [Consultado em 16/12/2017 às 19:00]

Disponível em <https://developer.android.com/reference/android/app/Service.html>

Tutorial Web [Consultado em 14/12/2017 às 19:10]

Disponível em <https://www.w3schools.com/>

Android Studio [Consultado em 09/12/2016 às 11:00]

Disponível em <https://developer.android.com/studio/index.html/>

Genymotion [Consultado em 09/12/2017 às 11:20]

Disponível em <https://www.genymotion.com/>

TERMO DE RESPONSABILIDADE de ORIENTAÇÃO

Eu, **JOÃO MANUEL DIAS DA SILVA**, grau: Doutor, declaro que o estudante, **GIDEÃO GIL LOPES ROCHA**, N.º 2843, Finalista do curso de Licenciatura em Informática de Gestão, realizou sob a minha orientação o Trabalho de Conclusão Curso, Projeto de Licenciatura intitulada: “**PHARM GO – APLICAÇÃO MÓVEL E PLATAFORMA WEB – INTERMEDIÇÃO COM FARMÁCIAS**”.

e que a mesma foi desenvolvida de acordo com as Normas de Elaboração e Apresentação dos TCC's da **UNIVERSIDADE DO MINDELO** e reúne todas as condições para a sua apresentação e defesa.

Mindelo, 20 de junho de 2018

O Orientador

